




## Gene expression

# CStreet: a computed Cell State trajectory inference method for time-series single-cell RNA sequencing data

Chengchen Zhao <sup>1,\*</sup>, Wenchao Xiu<sup>1,†</sup>, Yuwei Hua<sup>1,†</sup>, Naiqian Zhang <sup>2</sup> and Yong Zhang <sup>1,\*</sup>

<sup>1</sup>Institute for Regenerative Medicine, Shanghai East Hospital, Shanghai Key Laboratory of Signaling and Disease Research, Frontier Science Center for Stem Cell Research, School of Life Science and Technology, Tongji University, Shanghai 200092, China and <sup>2</sup>School of Mathematics and Statistics, Shandong University at Weihai, Weihai 264209, China

\*To whom correspondence should be addressed.

†The authors wish it to be known that, in their opinion, the first three authors should be regarded as Joint First Authors.

Associate Editor: Anthony Mathelier

Received on November 27, 2020; revised on June 24, 2021; editorial decision on June 28, 2021; accepted on June 30, 2021

## Abstract

**Motivation:** The increasing amount of time-series single-cell RNA sequencing (scRNA-seq) data raises the key issue of connecting cell states (i.e. cell clusters or cell types) to obtain the continuous temporal dynamics of transcription, which can highlight the unified biological mechanisms involved in cell state transitions. However, most existing trajectory methods are specifically designed for individual cells, so they can hardly meet the needs of accurately inferring the trajectory topology of the cell state, which usually contains cells assigned to different branches.

**Results:** Here, we present CStreet, a computed Cell State trajectory inference method for time-series scRNA-seq data. It uses time-series information to construct the  $k$ -nearest neighbor connections between cells within each time point and between adjacent time points. Then, CStreet estimates the connection probabilities of the cell states and visualizes the trajectory, which may include multiple starting points and paths, using a force-directed graph. By comparing the performance of CStreet with that of six commonly used cell state trajectory reconstruction methods on simulated data and real data, we demonstrate the high accuracy and high tolerance of CStreet.

**Availability and implementation:** CStreet is written in Python and freely available on the web at <https://github.com/TongjiZhanglab/CStreet> and <https://doi.org/10.5281/zenodo.4483205>

**Contact:** cczhao@tongji.edu.cn or yzhang@tongji.edu.cn

**Supplementary information:** [Supplementary data](#) are available at *Bioinformatics* online.

## 1 Introduction

Cell state transitions are the basis of the ontogenesis of multicellular organisms. In recent years, the increasing amount of time-series single-cell RNA sequencing (scRNA-seq) data has enabled transcriptional dynamics to be obtained, as well as the temporal transitions of cell states (Lederer and La Manno, 2020). However, it is practically impossible to obtain the transcriptome characteristics of one cell at multiple time points. Therefore, connecting cells with continuous cell states between adjacent time points has become a key issue in analyzing transcriptional dynamics over time (Griffiths *et al.*, 2018).

To address this issue, researchers have tried some alternatives in the experimental design and computational method development. At the experimental level, lineage tracing strategies that make use of the genetic recordings of cells have been used to trace the cell lineage changes at various time points (Wu *et al.*, 2019). However, these

methods have not yet been widely applied, and they cannot be applied to existing time-series scRNA-seq data without lineage tracing information. At the computational level, a series of single-cell data based trajectory inference methods such as TSCAN (Ji and Ji, 2016), Monocle 2 (Cao *et al.*, 2019; Qiu *et al.*, 2017; Trapnell *et al.*, 2014), PAGA (Wolf *et al.*, 2019), CytoTRACE (Gulati *et al.*, 2020), SCUBA (Marco *et al.*, 2014), TASIC (Rashid *et al.*, 2017), RNA velocity (La Manno *et al.*, 2018), Briggs's algorithm (Briggs *et al.*, 2018), STITCH (Wagner *et al.*, 2018), Waddington-OT (Schiebinger *et al.*, 2019), pseudodynamics (Fischer *et al.*, 2019), CSHMM (Lin and Bar-Joseph, 2019) and Tempora (Tran and Bader, 2020) have been developed (Saelens *et al.*, 2019; Sagar and Grun, 2020). Each of these computational methods has a specific design and worked well on meeting different needs. For example, Monocle constructed a minimum spanning tree (MST) based on transcriptome similarity to describe the cell state trajectory; CSHMM used continuous states HMM to reconstruct continuous

cell state trajectory; SCUBA modeled the development process using a stochastic dynamic system to identify bifurcation events; Tempora used biological pathway information to help identify cell type relationships; RNA velocity predicted the cell states in the near future by taking consideration of the intrinsic splicing kinetics. Although many of those methods incorporated time-series information to improve the performance of trajectory construction, most prior methods, such as CytoTRACE and RNA velocity, were specifically designed for individual cells. Since cells in one cell state may be inferred to have different branches of the trajectory, these single-cell based trajectory methods can hardly meet the needs of accurately inferring the topology of the cell state trajectory (Supplementary Fig. S1), which can highlight the unified biological mechanisms during cell state transitions.

Here, we present CStreet: a computed Cell State trajectory inference method for time-series scRNA-seq data. CStreet takes advantage of the time-series information of the input data to construct the  $k$ -nearest neighbors ( $k$ -NN) connections within each time point and between adjacent time points. Then, the cells in a cluster or of the same cell type are coarsely categorized to a cell state. CStreet uses a distribution-based parameter interval estimation to measure the transition probabilities of the cell states, while prior approaches used scoring, such as the percentages of votes used by Briggs *et al.* or the mutual information of the cluster pathway enrichment used by Tempora. The force-directed graph is created based on these connection probabilities to visualize the trajectory of the cell states, which may include multiple starting points and paths. By comparing the performance of CStreet with that of six commonly used cell state trajectory reconstruction methods on simulated data and real data, we demonstrate the high accuracy and high tolerance of CStreet.

## 2 Materials and methods

An expression matrix containing the time-series expression level as read counts or normalized values in tab-delimited or AnnData format (Wolf *et al.*, 2018) is accepted as the input of CStreet. Low-quality cell filtering and normalization can be conducted according to the input parameters. The cell state information can be inputted by the user or generated using the internal clustering function of CStreet. To assess and compare the performance of single-cell state trajectory inference methods, we simulated a series of scRNA-seq datasets using Splatter (Zappia *et al.*, 2017) and collected three time-series scRNA-seq datasets from ArrayExpress under accession E-MTAB-6967 (Pijuan-Sala *et al.*, 2019) and Gene Expression Omnibus under accession GSE90047 (Yang *et al.*, 2017) and GSE107122 (Yuzwa *et al.*, 2017). The expression matrices, measured as read counts, of Pijuan-Sala *et al.* (2019) and Yang *et al.* (2017) were downloaded from these public databases. The batch effects of these datasets were corrected using the *scanpy.pp.combat* function in SCANPY (Wolf *et al.*, 2018). The scaled expression matrix after normalization of Yuzwa *et al.* (2017) was downloaded from the supplementary information of Tran and Bader (2020). The scripts and the processed data can be found at the ‘Data&Code’ section on the <https://github.com/TongjiZhanglab/CStreet>.

### 2.1 Data simulation using splatter

We first used the *splatSimulatePaths* function in Splatter and generated a simulated scRNA-seq dataset with four continuous and partially overlapping cell paths that consisted of 12 000 cells and 2000 genes. Then, we used three time windows ( $t_1$ ,  $t_2$ ,  $t_3$ ), selecting 2000 cells from each to simulate the samplings of a time-series experiment. In this way, we obtained a simulated time-series scRNA-seq dataset containing two cell fate bifurcations of 2000 genes among 6000 cells. These cells can be divided into seven classes (one class at the  $t_1$  stage, i.e.  $C_{1A}$ ; two classes at the  $t_2$  stage, i.e.  $C_{2A}$  and  $C_{2B}$ ; and four classes at the  $t_3$  stage, i.e.  $C_{3A}$ ,  $C_{3B}$ ,  $C_{3C}$  and  $C_{3D}$ ). The known trajectories of these classes are  $C_{1A}$ – $C_{2A}$ ,  $C_{1A}$ – $C_{2B}$ ,  $C_{2A}$ – $C_{3A}$ ,  $C_{2A}$ – $C_{3B}$ ,  $C_{2B}$ – $C_{3C}$  and  $C_{2B}$ – $C_{3D}$ . We also simulated another two time-series datasets with different sampling densities by using five

time windows and seven time windows, each containing 1000 cells, on the same continuous paths.

### 2.2 Construction of $k$ -NN graphs using time-series scRNA-seq data

$k$ -NN graphs within each time point were constructed based on the Euclidean distance or the Pearson correlation coefficient of the cells in each cell state. Based on the  $k$ -NN graphs of individual cells, the cells in a cluster or of the same cell type are first coarsely categorized to a cell state; the interconnection numbers of cell states are calculated as the sum of cell interconnection numbers in the  $k$ -NN graphs. Then, connection probabilities between cell states are estimated by using these interconnection numbers. The force-directed graph is created based on these connection probabilities to visualize the layout of the cell states within each time point.

For the cells in adjacent time points, we find the  $k$ -nearest neighbors from the bidirections. A connection between two cells is introduced when they are identified as neighbors. The connections among these cells between adjacent time points are used in the subsequent estimation of the connection probabilities.

### 2.3 Calculation of connection probabilities of cell states

Under the assumption that each connection of cells is equally likely to occur, we denote event  $E$  as a connection between a cell in state  $A_x$  and a cell in state  $B_y$ . Assuming that the probability of the occurrence of  $E$  is  $p$ , then the probability of its nonoccurrence is  $q = 1 - p$ . Then, the probability distribution of the occurrence of  $E$  is assumed to follow a binomial distribution  $Bi(1, p)$ . To estimate the connection probability  $p$ , we adopt a strategy of repeated sampling trials using the frequency of cell connections. In each trial, a fraction of cells at the earlier time point ( $A_x$ ,  $M$  cells) or later time point ( $B_y$ ,  $N$  cells) is sampled to calculate the frequency ( $P_{A_x B_y}$ ) of the cell connections as follows:

$$P_{A_x B_y} = \frac{\sum_{i=1}^M \sum_{j=1}^N C_{a_i \& b_j}}{\sum_{i=1}^M C_{a_i} + \sum_{j=1}^N C_{b_j}}, \quad (1)$$

where  $C_{a_i \& b_j}$  denotes the number of connections between cell  $a_i$  in state  $A_x$  and cell  $b_j$  in state  $B_y$ .  $C_{a_i}$  denotes the times that cell  $a_i$  in state  $A_x$  was identified to connect to other cells and  $C_{b_j}$  denotes the times that cell  $b_j$  in state  $B_y$  was identified to connect to other cells. Then, the total number of the connections of the  $M$  cells in state  $A_x$  and the  $N$  cells in state  $B_y$  are summed. To ensure reliability, the mean of the frequency and its 95% confidence interval for 100 repeated trials is used as the estimated connection probability ( $p$ ). The connection threshold is determined by users or using Otsu’s method (Otsu, 1979), which calculates the optimum threshold separating the connected states and unconnected states. We assume that the cells at the later time point experienced a longer cell state transition than the cells at the earlier time point. Thus, the cell states between adjacent time points are chronologically directed along with the time series.

### 2.4 Expression matrix generation with rare cells and gene dropouts

To generate the expression matrix with different cell numbers, we randomly sample different numbers of cells ( $n = 3200, 1600, 800, 400, 200, 100$ ) from the previously described simulated dataset at the three time points. These downsampled datasets are then used to infer the cell state trajectory. For each number, the selection is performed 10 times with different random seeds.

To generate the expression matrix with different gene dropouts, we randomly sample fractions of 2000 genes to represent dropout rates from 10% to 60% (at intervals of 10% points) and changed their expression level to ‘not available (NA)’. For each fraction, the selection is performed 10 times with different random seeds.

## 2.5 Calculation of Hamming–Ipsen–Mikhailov (HIM) score and $F_1$ score

The HIM score is a combination of the Hamming distance and the Ipsen–Mikhailov distance to quantify the difference in the trajectory topologies. The Hamming distance measures the local structural similarities by calculating the distance between two graphs by matching individual edges in the adjacency matrix, and the Ipsen–Mikhailov distance measures the overall structural similarity by calculating the overall distance of two graphs based on matches between the adjacency matrix and its degree. Then, these clusters are connected based on the cell state trajectory results. The clusters with the same label are connected in both directions. In this way, we obtain the inferred trajectory topologies ( $T_{inf}$ ). The real trajectory topologies ( $T_{real}$ ) for the simulated data are recorded during the simulated data generation and for the real data are adopted from the related publications (Pijuan-Sala et al., 2019; Yang et al., 2017) and an embryonic development database (<https://discovery.lifemaps.com>). The HIM scores were calculated using the *nd.him* function in NetworkDistance (Jurman et al., 2011) to measure the difference between the inferred trajectory topologies and the real trajectory topologies according to the following formula:

$$\text{HIM}(T_{inf}, T_{real}) = \frac{\sqrt{H(T_{inf}, T_{real})^2 + \xi \times IM(T_{inf}, T_{real})^2}}{\sqrt{1 + \xi}}, \quad (2)$$

where  $H$  and  $IM$  represent the Hamming distance and the Ipsen–Mikhailov distance, respectively.  $\xi$  is a parameter to control the balance between these two distances and is estimated based on the number of nodes in the graph. For the comparisons of different methods in the same dataset, this parameter is consistent so that the HIM scores are comparable.

The  $F_1$  score is the harmonic mean of the precision and recall and measures the accuracy of a clustering scheme. In our case, this metric maps the cells in the real and predicted states by using their shared members. The precision is calculated as the ratio of the number of correctly predicted cells (true positives, TP) to the total number of predicted cells (the sum of TP and the number of false positives, FP). The recall is calculated as the ratio of the number of correctly predicted cells (TP) to the number of all real cells (the sum of TP and the number of false negatives, FN). Formally,

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad (3)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad (4)$$

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (5)$$

## 3 Results

### 3.1 Overview of CStreet

As shown in Figure 1, CStreet uses the expression matrix of time-series scRNA-seq data as input and constructs the cell state trajectory. It first builds  $k$ -NN graphs within and between the time points of cells that passed quality control and were normalized. Then, within each time point, the cells in a cluster or of the same cell type are coarsely categorized to a cell state with the number of interconnections calculated. Then connection probabilities between cell states are estimated by using these interconnection numbers. A force-directed graph (Jacomy et al., 2014) is created based on these connection probabilities to visualize the layout of the cell states within each time point and is used as the skeleton of the time-series trajectory. Next, we calculate the connection probabilities of each state at the earlier time point to the states at the later time point based on the  $k$ -NN graph between these two time points. Finally, based on the assumption that the cells at the later time point have experienced a longer cell state transition than the cells at the earlier time point, the cell states between adjacent time points were directed along with the time-series chronologically. The clustering results, connection probabilities and trajectory plot are outputted in separate files as well as an interactive states trajectory file that can be visualized using Cytoscape and its apps (Micale et al., 2014).

### 3.2 Evaluation on simulated datasets

To evaluate the performance of CStreet, we compared the trajectory accuracy and state assessment accuracy of CStreet with six commonly used cell state trajectory reconstruction methods (i.e. Monocle 2, TSCAN, SCUBA, PAGA, CSHMM and Tempora). As the pathway information, which is required by Tempora, is not applicable in simulated datasets, five commonly used methods were used on the simulated time-series scRNA-seq data (Fig. 2a and b; more description in Section 2). The HIM score was used to quantify the difference between the inferred trajectory and the ‘true trajectory’ (see more description in Section 2). The  $F_1$  score was used to measure the cell assessment accuracy. In this simulation dataset, CStreet and SCUBA accurately inferred all six trajectories among those cell states (Fig. 2c and f). Monocle 2 inferred the pseudotime of cells, and it failed to infer most trajectories (Fig. 2d). TSCAN constructed a cluster-based MST, and its main trajectory is  $C_{1A}-C_{1A}-C_{2A}-C_{3B}-C_{3D}-C_{2B}$ , in

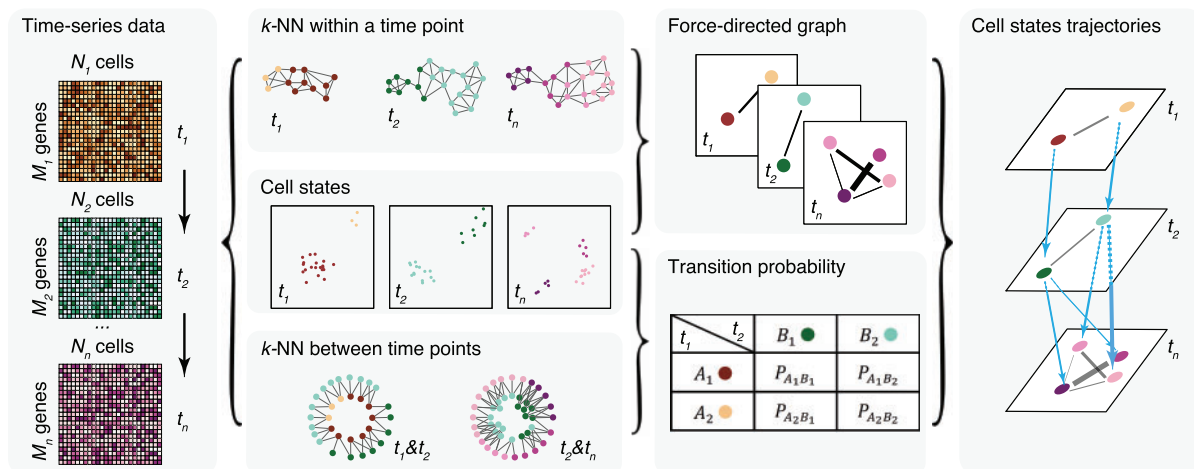
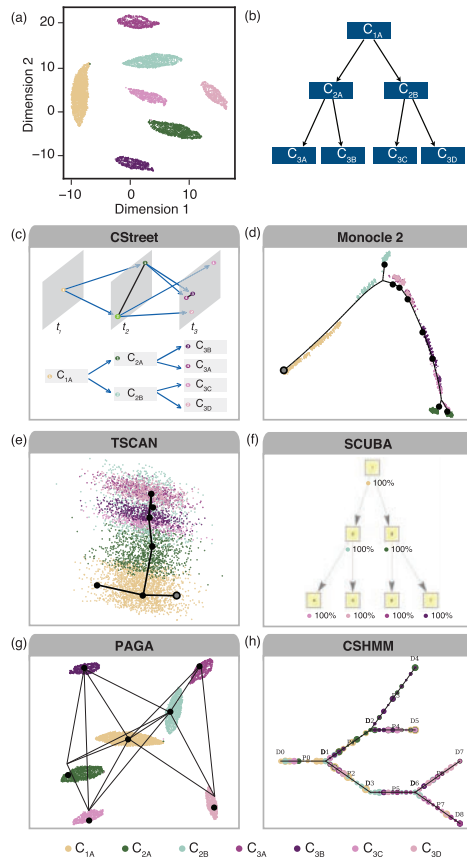


Fig. 1. The overall workflow of CStreet



**Fig. 2.** Comparison of CStreet with Monocle 2, TSCAN, SCUBA, PAGA and CSHMM on the simulated time-series data. (a) Scatter plot showing the visualization of the UMAP dimensional reduction output of the embryogenesis scRNA-seq data. Different cell states are plotted using different colors. (b) The true trajectory of the simulation data used to evaluate the accuracy of all the inferred trajectories. (c–h) The inferred cell state trajectories of the embryogenesis scRNA-seq data using CStreet (c), Monocle 2 (d), TSCAN (e), SCUBA (f), PAGA (g) and CSHMM (h). The black dots in (d), (e) and (g) represent the center of cell states and the gray dots in (d) and (e) represent the starts of trajectories. The black edges represent the connections of these centers. The thickness of edges in (c) and (g) represents the corresponding statistical measure of the connectivity between cell states. In (f), cell types and proportions contained in each cluster were labeled. In (h), each path represents a set of infinite states, each node represents the location where paths split, and each intermediate circle represents a cell state on the path. The circle sizes represent the numbers of cells assigned to the cell state. The cells are colored according to their true state

which  $C_{3B}-C_{3D}-C_{2B}$  were mistakenly identified (Fig. 2e). PAGA constructed a coarse-grained graph, which contained 14 connections, including 8 FP ones (Fig. 2g). In the result of CSHMM, the trajectories of  $C_{2B}-C_{3A}$ ,  $C_{2A}-C_{3C}$  and  $C_{3A}-C_{3D}$  were mistakenly identified, while  $C_{2A}-C_{3A}$ ,  $C_{2B}-C_{3C}$  and  $C_{2B}-C_{3D}$  were missed (Fig. 2h). The results based on simulated data showed that CStreet and SCUBA achieved the best performance in terms of trajectory inference and states assessment (Table 1). CStreet also

demonstrated reliable trajectory results using simulated datasets with different sampling densities (Supplementary Fig. S2).

For the real data with labeled cells, the calculation of the  $F_1$  scores was not applicable and only the HIM scores were calculated. For the simulated data that performed 10 times, the mean values of the HIM scores and  $F_1$  scores, as well as the 95% confidence intervals around the mean values, were calculated. HDTSD 1, mouse hepatoblast differentiation time-series data using the label strategy of intermediate cells; HDTSD 2, mouse hepatoblast differentiation time-series data using the label strategy of directed links; ETSD 1, mouse embryogenesis time-series data in the first three time points; ETSD 2, mouse embryogenesis time-series data in the all nine time points; CCTSD, early murine cerebral cortex development time-series data. NA means the result of the corresponding method is not available. The pathway information, which is required by Tempora, is not applicable in the simulated data. CSHMM failed to complete the trajectory construction in ETSD 1 and ETSD 2 datasets. Monocle 2, TSCAN, Tempora and SCUBA failed to complete the trajectory construction in ETSD 2 dataset.

### 3.3 Tolerance evaluation of cell rarity and gene dropouts

To further evaluate the tolerance of CStreet to rare cells and gene dropouts, we constructed trajectories using these methods on groups of simulation datasets with different numbers of cells and different numbers of detected genes. CStreet showed reliable results and the best trajectory inference accuracy and state assessment accuracy with 400 or more cells and gene dropout rates of 50% or lower (Fig. 3).

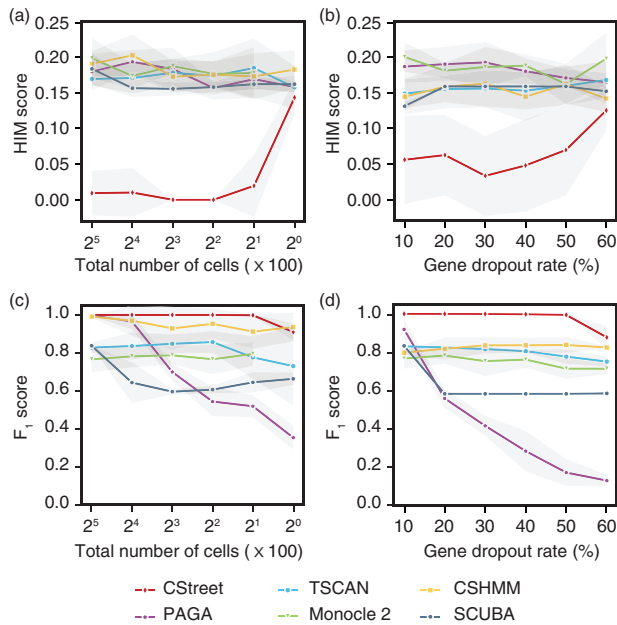
### 3.4 Application on real time-series scRNA-seq datasets

We next applied CStreet and the other methods on three real time-series scRNA-seq datasets with different characteristics, i.e. one having multiple sampling time points, one having multiple cell states and one having convergent cell state trajectories.

To evaluate the performance of CStreet on the dataset with multiple sampling time points, we compared the trajectory results of CStreet to the other six methods on a time-series scRNA-seq dataset at embryonic day 10.5 (E10.5, 54 cells), E11.5 (70 cells), E12.5 (41 cells), E13.5 (65 cells), E14.5 (70 cells), E15.5 (77 cells) and E17.5 (70 cells) during mouse hepatoblast differentiation. To distinguish the cell states for the inference of trajectories, we used two strategies to specifically label the cells with mixed cell states [the ‘hepatoblast/hepatocyte’ cells defined in Yang *et al.* (2017)]. The first strategy is labeling these cells in early time points (E10.5, E11.5) as ‘hepatoblast’, in late time points (E15.5, E17.5) as ‘hepatocyte’, and in intermediate time points (E12.5, E13.5, E14.5) as ‘intermediate cells’ (Fig. 4a and b). CStreet, TSCAN and SCUBA accurately inferred the transitions of hepatoblasts to intermediate cells, then to hepatocytes and cholangiocytes (Fig. 4c, Supplementary Fig. S3b and d). Tempora missed the trajectory of hepatoblasts to intermediate cells (Supplementary Fig. S3c); while PAGA and CSHMM mistakenly connected hepatocytes with cholangiocytes (Supplementary Fig. S3e and f). The second strategy is labeling these cells as hepatoblasts (E10.5, E11.5, E12.5 and E13.5) or hepatocytes (E14.5, E15.5 and E17.5) to construct a direct link from hepatoblasts to hepatocytes and cholangiocytes (Supplementary Fig. S4a and b).

**Table 1.** Comparison between CStreet and the other methods

Method	HIM score (simulated data)	$F_1$ score (simulated data)	HIM score (HDTSD 1)	HIM score (HDTSD 2)	HIM score (ETSD 1)	HIM score (ETSD 2)	HIM score (CCTSD)
CStreet	0.000 ± 0.000	1.000 ± 0.000	0.000	0.000	0.007	0.056	0.079
Monocle 2	0.209 ± 0.038	0.759 ± 0.033	0.397	0.636	0.065	NA	0.636
TSCAN	0.174 ± 0.034	0.817 ± 0.020	0.000	0.318	0.051	NA	0.397
Tempora	NA	NA	0.159	0.000	0.101	NA	0.079
SCUBA	0.000 ± 0.000	1.000 ± 0.000	0.000	0.318	0.058	NA	0.397
PAGA	0.185 ± 0.000	0.999 ± 0.000	0.556	0.636	0.101	0.033	0.477
CSHMM	0.124 ± 0.000	0.962 ± 0.000	0.238	0.318	NA	NA	0.477



**Fig. 3.** Performance comparison of CStreet and the other methods on the simulated datasets with different numbers of cells and different gene dropout rates. (a) Line plots showing the dynamic changes in the HIM score for the trajectory results using CStreet, TSCAN, CSHMM, PAGA, Monocle 2 and SCUBA on the simulated datasets with different numbers of cells. (b) Line plots showing the dynamic changes in the HIM score for the trajectory results on the simulated datasets with different gene dropout rates. (c) Line plots showing the dynamic changes in the F<sub>1</sub> score for the trajectory results on the simulated datasets with different numbers of cells. (d) Line plots showing the dynamic changes in the F<sub>1</sub> score for the trajectory results on simulated datasets with different gene dropout rates. The lines in the plot link the mean values of these scores. The gray shaded areas represent the 95% confidence intervals around the means

CStreet and Tempora correctly inferred the transition of hepatoblasts to hepatocytes and cholangiocytes (Supplementary Fig. S4c and f). TSCAN, SCUBA, PAGA and CSHMM mistakenly connected hepatocytes with cholangiocytes (Supplementary Fig. S4d, e and g–i). With both strategies, CStreet accurately inferred the trajectory of the cell states (Table 1), illustrating the applicability of CStreet to scRNA-seq datasets with multiple sampling time points.

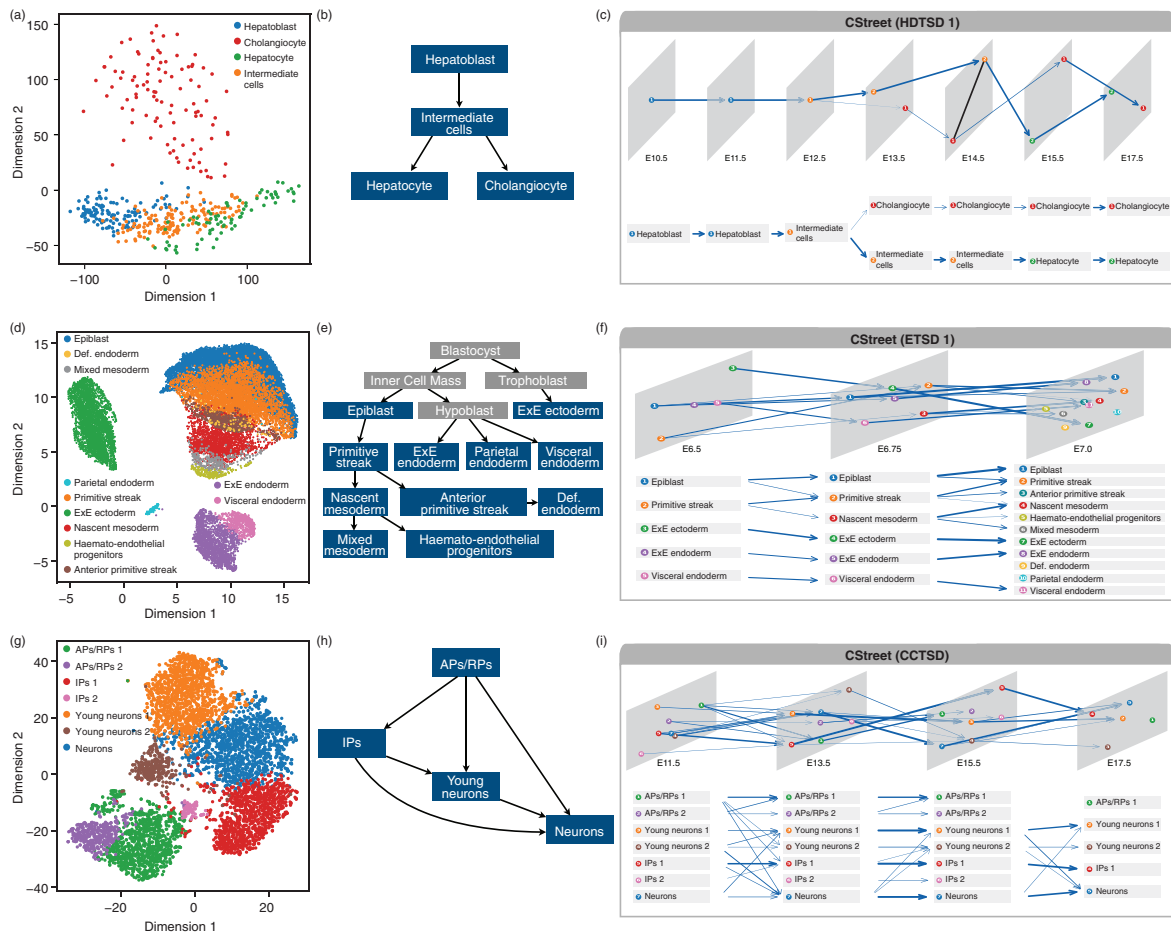
Next, to evaluate the performance of CStreet with multiple cell states at each time point, we constructed cell state trajectories using CStreet and other six methods at the first three time points (E6.5: 3482 cells, E6.75: 2067 cells and E7.0: 14 585 cells) of a time-series scRNA-seq dataset during mouse embryogenesis (Fig. 4d and e). CStreet accurately inferred the trajectories of epiblast to primitive streak, primitive streak to nascent mesoderm, together with independent trajectories of extraembryonic (ExE) endoderm, parietal endoderm and visceral endoderm. These independent trajectories might be caused by the cells in the earlier time point but having advanced developmental cell states. CStreet also correctly inferred the trajectory of primitive streak to anterior primitive streak, and the trajectory of nascent mesoderm to mixed mesoderm and haemato-endothelial progenitors, but it missed the trajectory of anterior primitive streak to definitive (Def.) endoderm (Fig. 4f). Monocle 2 correctly inferred the trajectory of epiblast to epiblast, but it mistakenly inferred the trajectories of haemato-endothelial progenitors to ExE ectoderm and ExE endoderm (Supplementary Fig. S5a). TSCAN correctly inferred the trajectories of epiblast to primitive streak and primitive streak to nascent mesoderm, but it mistakenly inferred the trajectories of primitive streak to epiblast, primitive streak to ExE ectoderm and ExE ectoderm to ExE endoderm (Supplementary Fig. S5b). Tempora correctly inferred the trajectory of anterior primitive streak to Def. endoderm, but it mistakenly inferred the trajectories of epiblast to visceral endoderm,

ExE endoderm to nascent mesoderm, parietal endoderm to mixed mesoderm and mixed mesoderm to ExE ectoderm (Supplementary Fig. S5c). SCUBA mistakenly inferred the trajectories of epiblast to ExE ectoderm and ExE ectoderm to ExE endoderm (Supplementary Fig. S5d). PAGA correctly inferred the connections of epiblast with primitive streak, primitive streak with nascent mesoderm and anterior primitive streak, but it mistakenly inferred the connections of visceral endoderm with ExE endoderm and epiblast with anterior primitive streak (Supplementary Fig. S5e). These results showed that CStreet outperformed other methods on this dataset (Table 1). To evaluate the performance of CStreet and other methods on complex trajectories, we further extended the comparisons using all the nine time points (E6.5, E6.75, E7.0, E7.25: 13 537 cells, E7.5: 10 994 cells, E7.75: 14 493 cells, E8.0: 16 681 cells, E8.25: 15 935 cells and E8.5: 16 909 cells) of the dataset (Supplementary Fig. S6a). As the dataset contained uncharacterized cell states, it was difficult to set a whole trajectory as the gold standard. Nevertheless, we curated reported trajectories between characterized cell states, i.e. a subset of the whole trajectory, to quantitatively evaluate the performance of the methods (Supplementary Fig. S6a and b). Both CStreet and PAGA showed comparable performance (Supplementary Fig. S6c and d; Table 1); while other methods failed to complete the trajectory construction, due to memory errors (Monocle 2, TSCAN, Tempora and SCUBA) and exceptionally excessive runtime (CSHMM). Taken together, these results demonstrated that CStreet is suitable to infer the trajectory of complex cell states.

Finally, we applied CStreet and other methods on a time-series scRNA-seq dataset with convergent trajectories, and we constructed cell state trajectories using CStreet and other six methods using data from E11.5 (1402 cells), E13.5 (1129 cells), E15.5 (2922 cells) and E17.5 (863 cells) during embryonic murine cerebral cortex development (Fig. 4g and h). CStreet and Tempora displayed better performance than other methods (Table 1). CStreet inferred all the correct cell state transitions, but it mistakenly inferred the trajectory of neurons to young neurons (Fig. 4i). Tempora accurately inferred most trajectories, but it missed the trajectory from intermediate progenitors (IPs) to neurons (Supplementary Fig. S7c). Monocle 2 mistakenly inferred trajectories of IPs to apical precursors/radial precursors (APs/RPs) and young neurons to IPs (Supplementary Fig. S7a). TSCAN inferred the correct trajectories of APs/RPs to IPs and IPs to neurons, but it mistakenly inferred trajectories of neurons to young neurons (Supplementary Fig. S7b). SCUBA inferred the correct trajectories of APs/RPs to neurons, APs/RPs to IPs and young neurons to neurons, but it mistakenly inferred trajectory of IPs to APs/RPs and neurons to young neurons (Supplementary Fig. S7d). PAGA inferred the correct trajectories of IPs to neurons and young neurons to neurons, but it mistakenly inferred trajectories of neurons to young neurons and young neurons to APs/RPs (Supplementary Fig. S7e). CSHMM inferred the correct trajectory of APs/RPs to neurons, but it mistakenly inferred the trajectory of neurons to young neurons (Supplementary Fig. S7f). These results illustrate the applicability of CStreet to scRNA-seq datasets having convergent cell states trajectories.

## 4 Discussion

The constructions of cell trajectories and cell state trajectories are two effective strategies to dissect the time-series scRNA-seq data. The construction of cell trajectories can demonstrate the relevance as well as the heterogeneity of each cell, while the construction of cell state trajectories can highlight the main path of cell state transitions and make the topology of the trajectory easier for interpretation. As mentioned in a trajectory inference method selection tool (Saelens et al., 2019), users need to take into account their specific needs, expected topology and the dataset dimensions to choose the most appropriate method to construct the trajectory. Each cell state trajectory inference method has a specific design. For example, Monocle 2 infers a graph manifold of single cells based on transcriptome similarity; instead of inferring cell state trajectory, the pseudotime ordering of cells is of greater concern when using



**Fig. 4.** Application of CStreet on three real time-series scRNA-seq datasets. (a) Scatter plot showing the visualization of the dimensional reduction output of the scRNA-seq data during mouse hepatoblast differentiation (HDTSD 1). Different states of cells are plotted using different colors. (b) The real trajectory of mouse hepatoblast differentiation with the label strategy of intermediate cells used to evaluate the accuracy of all the inferred trajectories. (c) The inferred cell state trajectory of CStreet on the scRNA-seq data during mouse hepatoblast differentiation using the label strategy of intermediate cells. (d) Scatter plot showing the visualization of the UMAP dimensional reduction output of the ETSD 1. Different cell states are plotted using different colors. (e) The real trajectory of ETSD 1 used to evaluate the accuracy of all the inferred trajectories. Blue boxes indicate cell states that are included in this dataset, while gray boxes indicate cell states that are not included. (f) The inferred cell state trajectory of CStreet on ETSD 1. (g) Scatter plot showing the visualization of the dimensional reduction output of the scRNA-seq data during murine cerebral cortex development (CCTSD). The coordinates and states of cells are obtained from Tran and Bader's paper. Different cell states are plotted using different colors. (h) The real trajectory of CCTSD used to evaluate the accuracy of all the inferred trajectories. (i) The inferred cell state trajectory of CStreet on the CCTSD. The thickness of edges in (c), (f) and (i) represents the estimated connection probabilities between cell states. ExE, extraembryonic; Def., definitive; APs/RPs, apical precursors and radial precursors; IPs, intermediate progenitors

Monocle 2. SCUBA starts with a binary tree model and uses a stochastic dynamic system to identify bifurcation events, and it performed well in the cases with typical bifurcation trajectories. Tempora uses biological pathway information to help identify cell type relationships, and it provides the significantly changed pathways along with the trajectory, which can benefit the users to explore novel biological insights of the trajectory. As a replenishment, CStreet provided another easy and accurate way to understand and interpret the trajectories of cell states. CStreet is designed for cell state trajectory construction using time-series scRNA-seq data. It obtains better, or at least comparable, cell state trajectory results compared to six commonly used cell state trajectory methods using both simulated data and real data. It also displays high tolerance to small cell numbers and high gene dropout rates. In addition, CStreet is available as a command line tool and a Python library to meet the different needs of users.

## Funding

This work has been supported by the National Key Research and Development Program of China [2017YFA0102600], National Natural Science Foundation of China [32030022, 31970642, 31721003, 31900491], National Program for Support of Top-notch Young

Professionals, China Postdoctoral Science Foundation [2018M642073], Major Program of Development Fund for Shanghai Zhangjiang National Innovation Demonstration Zone [ZJ2018-ZD-004].

*Conflict of Interest:* none declared.

## References

- Briggs, J.A. *et al.* (2018) The dynamics of gene expression in vertebrate embryogenesis at single-cell resolution. *Science*, **360**, eaar5780.
- Cao, J. *et al.* (2019) The single-cell transcriptional landscape of mammalian organogenesis. *Nature*, **566**, 496–502.
- Fischer, D.S. *et al.* (2019) Inferring population dynamics from single-cell RNA-sequencing time series data. *Nat. Biotechnol.*, **37**, 461–468.
- Griffiths, J.A. *et al.* (2018) Using single-cell genomics to understand developmental processes and cell fate decisions. *Mol. Syst. Biol.*, **14**, e8046.
- Gulati, G.S. *et al.* (2020) Single-cell transcriptional diversity is a hallmark of developmental potential. *Science*, **367**, 405–411.
- Jacomy, M. *et al.* (2014) ForceAtlas2, a continuous graph layout algorithm for handy network visualization designed for the Gephi software. *PLoS One*, **9**, e98679.
- Ji, Z. and Ji, H. (2016) TSCAN: pseudo-time reconstruction and evaluation in single-cell RNA-seq analysis. *Nucleic Acids Res.*, **44**, e117.

- Jurman, G. et al. (2011) An introduction to spectral distances in networks. *Front. Artif. Intel. Appl.*, **226**, 227–234.
- La Manno, G. et al. (2018) RNA velocity of single cells. *Nature*, **560**, 494–498.
- Lederer, A.R. and La Manno, G. (2020) The emergence and promise of single-cell temporal-omics approaches. *Curr. Opin. Biotechnol.*, **63**, 70–78.
- Lin, C. and Bar-Joseph, Z. (2019) Continuous-state HMMs for modeling time-series single-cell RNA-Seq data. *Bioinformatics*, **35**, 4707–4715.
- Marco, E. et al. (2014) Bifurcation analysis of single-cell gene expression data reveals epigenetic landscape. *Proc. Natl. Acad. Sci. USA*, **111**, E5643–E5650.
- Micale, G. et al. (2014) GASOLINE: a Cytoscape app for multiple local alignment of PPI networks. *F1000Res.*, **3**, 140.
- Otsu, N. (1979) A threshold selection method from gray-level histograms. *IEEE Trans. Syst. Man Cyber.*, **9**, 62–66.
- Pijuan-Sala, B. et al. (2019) A single-cell molecular map of mouse gastrulation and early organogenesis. *Nature*, **566**, 490–495.
- Qiu, X. et al. (2017) Reversed graph embedding resolves complex single-cell trajectories. *Nat. Methods*, **14**, 979–982.
- Rashid, S. et al. (2017) TASIC: determining branching models from time series single cell data. *Bioinformatics*, **33**, 2504–2512.
- Saelens, W. et al. (2019) A comparison of single-cell trajectory inference methods. *Nat. Biotechnol.*, **37**, 547–554.
- Sagar and Grun, D. (2020) Deciphering cell fate decision by integrated single-cell sequencing analysis. *Annu. Rev. Biomed. Data Sci.*, **3**, 1–22.
- Schiebinger, G. et al. (2019) Optimal-transport analysis of single-cell gene expression identifies developmental trajectories in reprogramming. *Cell*, **176**, 928–943.e922.
- Tran, T.N. and Bader, G.D. (2020) Tempora: cell trajectory inference using time-series single-cell RNA sequencing data. *PLoS Comput. Biol.*, **16**, e1008205.
- Trapnell, C. et al. (2014) The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nat. Biotechnol.*, **32**, 381–386.
- Wagner, D.E. et al. (2018) Single-cell mapping of gene expression landscapes and lineage in the zebrafish embryo. *Science*, **360**, 981–987.
- Wolf, F.A. et al. (2018) SCANPY: large-scale single-cell gene expression data analysis. *Genome Biol.*, **19**, 15.
- Wolf, F.A. et al. (2019) PAGA: graph abstraction reconciles clustering with trajectory inference through a topology preserving map of single cells. *Genome Biol.*, **20**, 59.
- Wu, S.S. et al. (2019) Lineage tracing: computational reconstruction goes beyond the limit of imaging. *Mol. Cells*, **42**, 104–112.
- Yang, L. et al. (2017) A single-cell transcriptomic analysis reveals precise pathways and regulatory mechanisms underlying hepatoblast differentiation. *Hepatology*, **66**, 1387–1401.
- Yuzwa, S.A. et al. (2017) Developmental emergence of adult neural stem cells as revealed by single-cell transcriptional profiling. *Cell Rep.*, **21**, 3970–3986.
- Zappia, L. et al. (2017) Splatter: simulation of single-cell RNA sequencing data. *Genome Biol.*, **18**, 174.